Matrix Factorization for Movie Recommendation Systems

Nipun Batra

IIT Gandhinagar

July 31, 2025

Outline

The Recommendation Problem

The Recommendation Problem

Think About It:

The Recommendation Problem

Think About It:

- You've rated 100 movies out of 15,000

The Recommendation Problem

Think About It:

- You've rated 100 movies out of 15,000
- Your friend has similar but different tastes

The Recommendation Problem

Think About It:

- You've rated 100 movies out of 15,000
- Your friend has similar but different tastes

The Recommendation Problem

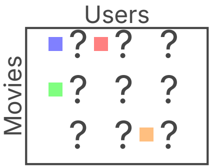
Think About It:

- You've rated 100 movies out of 15,000
- Your friend has similar but different tastes
- How do we predict what you'll like?

The Recommendation Problem

Think About It:

- You’ve rated 100 movies out of 15,000
- Your friend has similar but different tastes
- How do we predict what you’ll like?



Sparse Rating Matrix

!

The Magic Formula:

$$\text{Alice's rating} = \text{Alice's preferences} \cdot \text{Sholay's features}$$

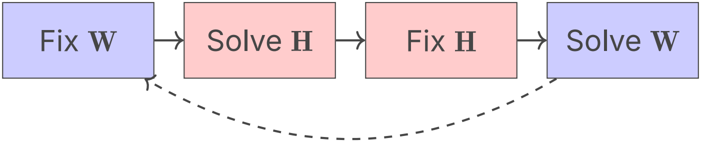
b Sholay's features: $\mathbf{h}_1 = [0.95, 0.10, 0.85]^T$

$$\hat{a}_{11} = \mathbf{w}_1^T \mathbf{h}_1 \tag{1}$$
$$= w_{11} \cdot 0.95 + w_{12} \cdot 0.10 + w_{13} \cdot 0.85 \tag{2}$$

Goal: Find w_{11}, w_{12}, w_{13} such that $\hat{a}_{11} \approx 5$ (Alice's actual rating)
[=

1. $\mathbf{A} \in \mathbb{R}^{N \times M}$
2. $\mathbf{W} \in \mathbb{R}^{N \times r}$
3. $\mathbf{H} \in \mathbb{R}^{r \times M}$
4. Total parameters: $Nr + rM = r(N + M)$

bInitialize: $\mathbf{W}^{(0)}$ and $\mathbf{H}^{(0)}$ randomly



$$\text{minimize}_{\mathbf{w}_i} \sum_{j:(i,j)\in\Omega} (a_{ij} - \mathbf{w}_i^T \mathbf{h}_j)^2$$

$$\text{minimize}_{\mathbf{w}_i} \sum_{j:(i,j)\in\Omega} (a_{ij} - \mathbf{w}_i^T \mathbf{h}_j)^2$$

Matrix Form for User *i*: Let $\Omega_i = \{j : (i,j) \in \Omega\}$ (movies rated by user *i*)

$$\text{minimize}_{\mathbf{w}_i} \sum_{j:(i,j)\in\Omega} (a_{ij} - \mathbf{w}_i^T \mathbf{h}_j)^2$$

Matrix Form for User *i*: Let $\Omega_i = \{j : (i,j) \in \Omega\}$ (movies rated by user *i*)

$$\mathbf{y}_i = [a_{i,j_1}, a_{i,j_2}, \ldots, a_{i,j_{|\Omega_i|}}]^T \tag{3}$$

$$\mathbf{X}_i = [\mathbf{h}_{j_1}, \mathbf{h}_{j_2}, \ldots, \mathbf{h}_{j_{|\Omega_i|}}]^T \tag{4}$$

$$\text{minimize}_{\mathbf{w}_i} \sum_{j:(i,j)\in\Omega} (a_{ij} - \mathbf{w}_i^T \mathbf{h}_j)^2$$

Matrix Form for User *i*: Let $\Omega_i = \{j : (i,j) \in \Omega\}$ (movies rated by user *i*)

$$\mathbf{y}_i = [a_{i,j_1}, a_{i,j_2}, \ldots, a_{i,j_{|\Omega_i|}}]^T \quad (3)$$

$$\mathbf{X}_i = [\mathbf{h}_{j_1}, \mathbf{h}_{j_2}, \ldots, \mathbf{h}_{j_{|\Omega_i|}}]^T \quad (4)$$

Least Squares Solution:

$$\boxed{\mathbf{w}_i^* = (\mathbf{X}_i^T \mathbf{X}_i)^{-1} \mathbf{X}_i^T \mathbf{y}_i}$$

$$\mathbf{y}_1 = \begin{bmatrix} 5 \\ 4 \\ 2 \\ 3 \\ 2 \end{bmatrix} \tag{5}$$

$$\mathbf{X}_1 = \begin{bmatrix} 0.95 & 0.10 & 0.85 \\ 1.00 & 0.20 & 0.90 \\ 0.05 & 0.80 & 0.30 \\ 0.05 & 0.95 & 0.70 \\ 0.05 & 0.15 & 0.95 \end{bmatrix} \tag{6}$$

$$\mathbf{y}_1 = \begin{bmatrix} 5 \\ 4 \\ 2 \\ 3 \\ 2 \end{bmatrix} \tag{5}$$

$$\mathbf{X}_1 = \begin{bmatrix} 0.95 & 0.10 & 0.85 \\ 1.00 & 0.20 & 0.90 \\ 0.05 & 0.80 & 0.30 \\ 0.05 & 0.95 & 0.70 \\ 0.05 & 0.15 & 0.95 \end{bmatrix} \tag{6}$$

Solution: $\mathbf{w}_1^* = (\mathbf{X}_1^T \mathbf{X}_1)^{-1} \mathbf{X}_1^T \mathbf{y}_1$
This gives us Alice's feature preferences!

$$\text{minimize}_{\mathbf{h}_j} \sum_{i:(i,j)\in\Omega} (a_{ij} - \mathbf{w}_i^T \mathbf{h}_j)^2$$

$$\text{minimize}_{\mathbf{h}_j} \sum_{i:(i,j)\in\Omega} (a_{ij} - \mathbf{w}_i^T \mathbf{h}_j)^2$$

Matrix Form for Movie j: Let $\Omega_j = \{i : (i,j) \in \Omega\}$ (users who rated movie j)

$$\text{minimize}_{\mathbf{h}_j} \sum_{i:(i,j)\in\Omega} (a_{ij} - \mathbf{w}_i^T \mathbf{h}_j)^2$$

Matrix Form for Movie j: Let $\Omega_j = \{i : (i,j) \in \Omega\}$ (users who rated movie j)

$$\mathbf{y}_j = [a_{i_1,j}, a_{i_2,j}, \ldots, a_{i_{|\Omega_j|},j}]^T \quad (7)$$

$$\mathbf{X}_j = [\mathbf{w}_{i_1}, \mathbf{w}_{i_2}, \ldots, \mathbf{w}_{i_{|\Omega_j|}}]^T \quad (8)$$

$$\text{minimize}_{\mathbf{h}_j} \sum_{i:(i,j)\in\Omega} (a_{ij} - \mathbf{w}_i^T \mathbf{h}_j)^2$$

Matrix Form for Movie j: Let $\Omega_j = \{i : (i,j) \in \Omega\}$ (users who rated movie j)

$$\mathbf{y}_j = [a_{i_1,j}, a_{i_2,j}, \ldots, a_{i_{|\Omega_j|},j}]^T \tag{7}$$

$$\mathbf{X}_j = [\mathbf{w}_{i_1}, \mathbf{w}_{i_2}, \ldots, \mathbf{w}_{i_{|\Omega_j|}}]^T \tag{8}$$

Least Squares Solution:

$$\boxed{\mathbf{h}_j^* = (\mathbf{X}_j^T \mathbf{X}_j)^{-1} \mathbf{X}_j^T \mathbf{y}_j}$$

Objective Function:

$$L(\mathbf{W}, \mathbf{H}) = \sum_{(i,j) \in \Omega} (a_{ij} - \mathbf{w}_i^T \mathbf{h}_j)^2$$

Objective Function:

$$L(\mathbf{W}, \mathbf{H}) = \sum_{(i,j)\in\Omega} (a_{ij} - \mathbf{w}_i^T \mathbf{h}_j)^2$$

Gradients:

$$\frac{\partial L}{\partial \mathbf{w}_i} = -2 \sum_{j:(i,j)\in\Omega} (a_{ij} - \mathbf{w}_i^T \mathbf{h}_j)\mathbf{h}_j \tag{9}$$

$$\frac{\partial L}{\partial \mathbf{h}_j} = -2 \sum_{i:(i,j)\in\Omega} (a_{ij} - \mathbf{w}_i^T \mathbf{h}_j)\mathbf{w}_i \tag{10}$$

Your Process:

Your Process:

1. Make a guess about their rating

Your Process:

1. Make a guess about their rating
2. See their actual rating

Your Process:

1. Make a guess about their rating
2. See their actual rating
3. Adjust your understanding

Your Process:

1. Make a guess about their rating
2. See their actual rating
3. Adjust your understanding
4. Repeat for next movie

Your Process:

1. Make a guess about their rating
2. See their actual rating
3. Adjust your understanding
4. Repeat for next movie

SGD does exactly this!

Your Process:

1. Make a guess about their rating
2. See their actual rating
3. Adjust your understanding
4. Repeat for next movie

SGD does exactly this!

- One rating at a time

Your Process:

1. Make a guess about their rating
2. See their actual rating
3. Adjust your understanding
4. Repeat for next movie

SGD does exactly this!

- One rating at a time
- Small adjustments

Your Process:

1. Make a guess about their rating
2. See their actual rating
3. Adjust your understanding
4. Repeat for next movie

SGD does exactly this!

- One rating at a time
- Small adjustments

Your Process:

1. Make a guess about their rating
2. See their actual rating
3. Adjust your understanding
4. Repeat for next movie

SGD does exactly this!

- One rating at a time
- Small adjustments
- Gradually improves

1. **Predict:** $\hat{a}_{ij} = \mathbf{w}_i^T \mathbf{h}_j$

1. **Predict:** $\hat{a}_{ij} = \mathbf{w}_i^T \mathbf{h}_j$
2. **Compute Error:** $e_{ij} = a_{ij} - \hat{a}_{ij}$

1. **Predict:** $\hat{a}_{ij} = \mathbf{w}_i^T \mathbf{h}_j$
2. **Compute Error:** $e_{ij} = a_{ij} - \hat{a}_{ij}$
3. **Update:**

$$\mathbf{w}_i \leftarrow \mathbf{w}_i + \alpha \cdot e_{ij} \cdot \mathbf{h}_j \tag{11}$$

$$\mathbf{h}_j \leftarrow \mathbf{h}_j + \alpha \cdot e_{ij} \cdot \mathbf{w}_i \tag{12}$$

1. **Predict:** $\hat{a}_{ij} = \mathbf{w}_i^T \mathbf{h}_j$
2. **Compute Error:** $e_{ij} = a_{ij} - \hat{a}_{ij}$
3. **Update:**

$$\mathbf{w}_i \leftarrow \mathbf{w}_i + \alpha \cdot e_{ij} \cdot \mathbf{h}_j \tag{11}$$

$$\mathbf{h}_j \leftarrow \mathbf{h}_j + \alpha \cdot e_{ij} \cdot \mathbf{w}_i \tag{12}$$

1. **Predict:** $\hat{a}_{ij} = \mathbf{w}_i^T \mathbf{h}_j$
2. **Compute Error:** $e_{ij} = a_{ij} - \hat{a}_{ij}$
3. **Update:**

$$\mathbf{w}_i \leftarrow \mathbf{w}_i + \alpha \cdot e_{ij} \cdot \mathbf{h}_j \tag{11}$$

$$\mathbf{h}_j \leftarrow \mathbf{h}_j + \alpha \cdot e_{ij} \cdot \mathbf{w}_i \tag{12}$$

Intuition:

- If $e_{ij} > 0$: Predicted rating too low → Increase similarity

1. **Predict:** $\hat{a}_{ij} = \mathbf{w}_i^T \mathbf{h}_j$
2. **Compute Error:** $e_{ij} = a_{ij} - \hat{a}_{ij}$
3. **Update:**

$$\mathbf{w}_i \leftarrow \mathbf{w}_i + \alpha \cdot e_{ij} \cdot \mathbf{h}_j \tag{11}$$

$$\mathbf{h}_j \leftarrow \mathbf{h}_j + \alpha \cdot e_{ij} \cdot \mathbf{w}_i \tag{12}$$

Intuition:

- If $e_{ij} > 0$: Predicted rating too low → Increase similarity
- If $e_{ij} < 0$: Predicted rating too high → Decrease similarity

1. **Predict:** $\hat{a}_{ij} = \mathbf{w}_i^T \mathbf{h}_j$
2. **Compute Error:** $e_{ij} = a_{ij} - \hat{a}_{ij}$
3. **Update:**

$$\mathbf{w}_i \leftarrow \mathbf{w}_i + \alpha \cdot e_{ij} \cdot \mathbf{h}_j \tag{11}$$

$$\mathbf{h}_j \leftarrow \mathbf{h}_j + \alpha \cdot e_{ij} \cdot \mathbf{w}_i \tag{12}$$

Intuition:

- If $e_{ij} > 0$: Predicted rating too low → Increase similarity
- If $e_{ij} < 0$: Predicted rating too high → Decrease similarity

1. **Predict:** $\hat{a}_{ij} = \mathbf{w}_i^T \mathbf{h}_j$
2. **Compute Error:** $e_{ij} = a_{ij} - \hat{a}_{ij}$
3. **Update:**

$$\mathbf{w}_i \leftarrow \mathbf{w}_i + \alpha \cdot e_{ij} \cdot \mathbf{h}_j \tag{11}$$

$$\mathbf{h}_j \leftarrow \mathbf{h}_j + \alpha \cdot e_{ij} \cdot \mathbf{w}_i \tag{12}$$

Intuition:

- If $e_{ij} > 0$: Predicted rating too low → Increase similarity
- If $e_{ij} < 0$: Predicted rating too high → Decrease similarity
- Learning rate α controls step size

$$\text{Current: } \mathbf{w}_1 = [0.4, 0.2, 0.3], \quad \mathbf{h}_1 = [0.95, 0.10, 0.85] \tag{13}$$

$$\text{Prediction: } \hat{a}_{11} = 0.4 \times 0.95 + 0.2 \times 0.10 + 0.3 \times 0.85 = 0.655 \tag{14}$$

$$\text{Error: } e_{11} = 5 - 0.655 = 4.345 \tag{15}$$

$$\text{Current: } \mathbf{w}_1 = [0.4, 0.2, 0.3], \quad \mathbf{h}_1 = [0.95, 0.10, 0.85] \tag{13}$$

$$\text{Prediction: } \hat{a}_{11} = 0.4 \times 0.95 + 0.2 \times 0.10 + 0.3 \times 0.85 = 0.655 \tag{14}$$

$$\text{Error: } e_{11} = 5 - 0.655 = 4.345 \tag{15}$$

Updates with $\alpha = 0.01$:

$$\mathbf{w}_1 \leftarrow [0.4, 0.2, 0.3] + 0.01 \times 4.345 \times [0.95, 0.10, 0.85] \tag{16}$$

$$= [0.4413, 0.2043, 0.3369] \tag{17}$$

$$\mathbf{h}_1 \leftarrow [0.95, 0.10, 0.85] + 0.01 \times 4.345 \times [0.4, 0.2, 0.3] \tag{18}$$

$$= [0.9674, 0.1087, 0.8631] \tag{19}$$

!

Answers:

b

$X_1 = \begin{bmatrix} 1.0 & 0.3 \\ 0.2 & 0.8 \end{bmatrix}$ (columns 1,3 of $\mathbf{H}^{(0)T}$)

Solve: $\mathbf{w}_1^{(1)} = (\mathbf{X}_1^T\mathbf{X}_1)^{-1}\mathbf{X}_1^T\mathbf{y}_1$

Continue for all users and movies...

[=

- Distributed computing, approximate algorithms, caching batrix factorization can only work with explicit ratings

Answers:

stability requirements